

What is Claimed Is:

1. A method of communicating between first and second machines, said method comprising the steps of:

generating a message at a first machine including at least one argument and a type label for said argument; and
transmitting said message from said first machine.

2. A method according to claim 1, wherein said message is a service invocation request for invoking a service at said second machine.

3. A method of invoking a service at a first machine, said method comprising the steps of:

receiving at said first machine a service invocation request generated at a second machine, said service invocation request including at least one argument and a type label for said argument; and
invoking said service in response to said request.

4. A method according to claim 2 or 3, wherein said service invocation request further includes a semantic label for said argument.

5. A method according to claim 2 or 3, wherein said service invocation request includes multiple arguments and includes a type label for all arguments in said request.

6. A method according to claim 5, wherein said service invocation request includes semantic labels for all arguments in said request.

7. A method according to claim 5, wherein said service invocation request contains multiple data items and said request further includes semantic labels for all data items in said request.

8. A method of communicating between first and second machines, said method comprising the steps of:

generating a message at said second machine, said message including at least one argument containing at least one data item, said message further including a semantic label for said data item; and

transmitting said message from said second machine.

9. A method according to claim 8, wherein said message is a service invocation request for invoking a service at said first machine.

10. A method of invoking a service at a first machine, said method comprising the steps of:

receiving at said first machine a service invocation request generated at a second machine, said request including at least one argument containing at least one data item, said service invocation request further including a semantic label for said data item; and

invoking said service in response to said request.

11. A method according to claim 9 or 10, wherein said service invocation request further includes a type label for said data item.

12. A method according to claim 9 or 10, wherein said request contains multiple data items and wherein said request includes semantic labels for all data items contained in said request.

13. A method according to either of claims 9 or 10, wherein said request further includes at least one further argument for which a semantic label is provided but for which semantic labels are not provided for all data items in said further argument.

transmitting from said first machine a service invocation reply including at least one output argument and a type label for said argument.

23. A method of invoking a service at a first machine, said method comprising the steps of:

transmitting a service invocation request from a second machine; and
receiving at said second machine a service invocation reply including at least one output argument and a type label for said argument.

24. A method according to claim 22 or 23, wherein said service invocation reply further includes a semantic label for said output argument.

25. A method according to claim 22 or 23, wherein said service invocation reply includes multiple output arguments and includes a type label for all output arguments in said reply.

26. A method according to claim 24, wherein said service invocation reply includes semantic labels for all output arguments in said reply.

27. A method according to claim 22 or 23, wherein said service invocation reply includes multiple data items and said reply further includes a type label for all data items in said reply.

28. A method of invoking a service at a first machine, said method comprising the steps of:

receiving at said first machine a service invocation request ;
invoking said service in response to said request; and
transmitting from said first machine a service invocation reply including at least one output argument containing at least one data item, said service invocation reply further including a semantic label for said data item.

29. A method of invoking a service at a first machine, said method comprising the steps of:

transmitting a service invocation request from a second machine; and
receiving at the second machine a service invocation reply including at least one output argument containing at least one data item, said service invocation reply further including a semantic label for said data item.

30. A method according to claim 28 or 29, wherein said service invocation reply further includes a type label for said data item.

31. A method according to claim 28 or 29, wherein said reply contains multiple data items and wherein said reply includes semantic labels for all data items contained in said reply.

32. A method according to claim 28 or 29, wherein said reply further includes at least one further argument for which a semantic label is provided but for which semantic labels are not provided for all data items in said further argument.

33. A method according to any one of claims 23, 24, 28 or 29 wherein said first and second machines are networked computers.

34. A method according to any one of claims 23 or 29, wherein said request is expressed in a markup language.

35. A method according to claim 22, wherein said request is expressed in a markup language.

36. A method according to any one of claims 23, 24, 28 or 29, wherein said service comprises a function.

37. A method according to claim 38, wherein said service comprises a function.

38. A method according to claims 37, wherein said service invocation reply is in the form of an XML document.

39. A method according to any one of claims 23, 24, 28 or 29, wherein said service invocation reply is in the form of an XML document.

40. A method according to claim 33, wherein said service invocation reply is transmitted using a transfer protocol selected from the group consisting of HTTP, FTP and SMTP.

41. A method according to any one of claims 23, 24, 28 or 29, wherein said service invocation reply is transmitted using a transfer protocol selected from the group consisting of HTTP, FTP and SMTP.

42. A method according to claim 40, wherein said transfer protocol includes a header, and said request identifies said service to be invoked by including a service name in said header.

43. A method according to claim 40, wherein said transfer protocol includes a destination indicator, and said request identifies said service to be invoked by including a service name in said destination indicator.

44. A method according to claim 43, wherein said destination indicator comprises a URL or URI.

45. A method according to claim 18, wherein said service is identified by the element type name of the root element of the XML document.

46. A method according to claim 18, wherein said service is identified as a function of the XML document type specified for said XML document.

47. A method according to claim 46, wherein said service is a particular service corresponding to the XML document type specified for said XML document.

48. A method according to claim 18, wherein said first machine accepts only XML documents of a particular document type corresponding to the service to be invoked.

49. A method of invoking a service at a first machine, said method comprising the steps of:

generating at a second machine a service invocation request expressed in a markup language; and
transmitting said message from said second machine.

50. A method of invoking a service at a first machine, said method comprising the steps of:

receiving at said first machine a service invocation request generated at a second machine, said service invocation request expressed in a markup language; and
invoking said service in response to said request.

51. A method according to claim 49 or 50, wherein said service invocation request is transmitted using an HTTP transfer protocol.

52. A method according to claim 51, wherein said first and second machines are networked computers.

53. A method according to claim 52, wherein said service comprises a function.

54. A method according to claim 49 or 50, wherein said service invocation request is an XML document.

55. A method of invoking a service at a first machine, said method comprising the steps of:

receiving at said first machine a service invocation request ;

invoking said service in response to said request; and

transmitting from said first machine a service invocation reply expressed in a markup language.

56. A method of invoking a service at a first machine, said method comprising the steps of:

transmitting a service invocation request from a second machine; and

receiving at said second machine a service invocation reply expressed in a markup language.

57. A method according to claim 55 or 56, wherein said service invocation reply is transmitted using a transfer protocol selected from the group consisting of HTTP, FTP and SMTP.

58. A method according to claim 57, wherein said first and second machines are networked computers.

59. A method according to claim 58, wherein said service comprises a function.

60. A method according to claim 52 or 58, wherein said service is a database query.

61. A method according to claim 52 or 58, wherein said service is a web site.

62. A method according to claim 52 or 58, wherein said service is a middleware platform.

63. A method according to claim 57, wherein said service invocation request is an XML document.

64. A method of invoking a service at a first machine from a second machine, comprising the steps of generating a service invocation request at said second machine in compliance with a markup language-based message encoding, and transmitting said service invocation request from said second machine, said message including plural elements and wherein all elements in said message have element type names selected from an encoding group consisting of no more than six element type names.

65. A method of invoking a service at a first machine, comprising the steps of:

receiving at said first machine a service invocation request generated at a second machine in compliance with a markup language-based message encoding, said message including plural elements and wherein all elements in said message have element type names selected from an encoding group consisting of no more than six element type names; and

invoking said service in response to said request.

66. A method according to claim 64 or 65, wherein said encoding group comprises at least a first element (e.g., VALUE) for representing data, a second and third (e.g., RECORD and either ARRAY or LIST) each for containing a set of elements belonging to said group.

67. A method according to claim 66, wherein said group includes a fourth element type name (e.g., OBJECT) for designating an element specifying a reference uniquely identifying another element in said service invocation request.

68. A method according to claim 66, wherein said group includes a further element type name (e.g., NULL) for designating an element representing the absence of a data item.

69. A method according to claim 66, wherein one of said third and fourth element type names (e.g., ARRAY) designates an element containing an array of elements having element type names belonging to said group.

70. A method of invoking a service at a first machine from a second machine, comprising the steps of generating a service invocation request at said second machine using a markup language-based message encoding, and transmitting said service invocation request from said second machine, wherein said message includes plural elements and wherein all of said elements have element type names selected from an encoding group having a predetermined number of members, with at least two of said members (e.g., RECORD and ARRAY) designating elements containing other elements having element type names belonging to said group.

71. A method of invoking a service at a first machine, comprising the steps of:

receiving at said first machine a service invocation request generated at a second machine in compliance with a markup language-based message encoding, wherein said message includes plural elements and wherein all elements in said message have element type names selected from an encoding group having a predetermined number of members, with at least two of said members (e.g., RECORD and ARRAY) designating elements containing other elements having element type names belonging to said group; and

invoking said service in response to said request.

72. A method of invoking a service at a first machine from a second machine, comprising the steps of:

generating a service invocation request message at said second machine in compliance with a markup language-based message encoding, wherein said message includes plural elements and wherein all elements in said message have element type names selected from an encoding group having a predetermined number of element type names, including at least a first (e.g., VALUE) element type name for designating an element containing data, and a second (e.g., RECORD) element type name for designating an element containing a set of children elements having element type names selected from said group; and

transmitting said message .

73. A method of invoking a service at a first machine, comprising the steps of:

receiving at said first machine a service invocation request message generated at a second machine in compliance with a markup language-based message encoding, wherein said message includes plural elements and wherein all elements in said message have element type names selected from an encoding group having a predetermined number of element type names, including at least a first (e.g., VALUE) element type name for designating an element containing data, and a second (e.g., RECORD) element type name for designating an element containing a set of children elements having element type names selected from said group; and

invoking said service in response to said message.

74. A method of invoking a service at a first machine, said method comprising the steps of:

receiving at said first machine a service invocation request;

invoking said service in response to said request; and

transmitting from said first machine a service invocation reply message in compliance with a markup language-based message encoding, wherein said

message includes plural elements and wherein all elements in said message have element type names selected from an encoding group having a predetermined number of element type names, including at least a first (e.g., VALUE) element type name for designating an element containing data, and a second (e.g., RECORD) element type name for designating an element containing a set of children elements having element type names selected from said group.

75. A method of invoking a service at a first machine, said method comprising the steps of:

transmitting a service invocation request from a second machine; and

receiving at said second machine a service invocation reply message in compliance with a markup language-based message encoding, wherein said message includes plural elements and wherein all elements in said message have element type names selected from an encoding group having a predetermined number of element type names, including at least a first (e.g., VALUE) element type name for designating an element containing data, and a second (e.g., RECORD) element type name for designating an element containing a set of children elements having element type names selected from said group.

76. A method according to claim any one of claims 72-75, wherein said encoding group further includes a third element type name (e.g., LIST or ARRAY) for designating an element containing a set of elements having element type names selected from said group.

77. A method according to claim 76, wherein said encoding group includes a fourth element type name (e.g., ARRAY or LIST) for designating an element containing a set of elements having element type names selected from said group.

78. A method according to claim 76, wherein said encoding group includes a fourth element type name (e.g., OBJECT) for designating an element uniquely identifying another encoding element within a particular message.

79. A method according to claim 76, wherein said encoding group includes a fourth element type name (e.g., NULL) for designating the absence of a data item.

80. A method according to claim 78, wherein said encoding group includes a fifth element type name (e.g., NULL) for designating the absence of a data item.

81. A method according to claim 80, wherein said encoding group includes a sixth element type name (e.g., ARRAY or LIST) for designating an element containing a set of elements having element type names selected from said group.

82. A method according to claim 76, wherein said third element type name (e.g., ARRAY) designates an element containing an n-dimensional array (where n is an integer such that $n \geq 1$) of elements having element type names selected from said encoding group.

83. A method according to any one of claims 72-75, wherein said encoding provides a type label associated with an element having said first element type name.

84. A method according to claim 83, wherein an element of said first element type name with no type label is assumed to be a string type element.

85. A method according to claim 83, wherein said type label is expressed as an XML attribute on said element having said first element type name, with the data type of a data item contained in said element is designated by the value of said attribute.

86. A method according to claim 76, wherein said encoding group further includes a fourth element type name (e.g., NUMBER) for designating an element representing a numeric value.

87. A method according to claim 76, wherein said encoding group includes multiple type names each designating a respective different type of data item contained in an element having said first type name.

88. A method according to claim 76, wherein said message further includes a semantic label for at least one data item contained in said message.

89. A method according to claim 88, wherein said semantic label is represented by the value of an XML attribute on the element containing said data item.

90. A method of invoking a service at a first machine from a second machine, said method comprising the steps of:

generating a service invocation request message at said second machine in compliance with a markup language-based message encoding wherein each element in said message is associated with a type indicator selected from a group of names, said group of names including at least an array type name indicating that the corresponding element is an n-dimensional array containing a plurality of data items, where n is an integer and $n \geq 1$, said message including at least one data item which is a multi-level nested array element where each nesting level corresponds to a respective dimension of said array element; and

transmitting said service invocation request message from said second machine.

91. A method of invoking a service at a first machine, comprising the steps of:

receiving at said first machine a service invocation request message generated at a second machine in compliance with a markup language-based message encoding wherein each element in said message is associated with a type indicator selected from a group of names, said group of names including at least an array type name indicating that the corresponding element is an n-dimensional array containing a plurality of data items, where n is an integer and $n \geq 1$, said message including at least one data item which is a multi-level nested array element where each nesting level corresponds to a respective dimension of said array element; and

invoking said service in response to said message.

92. A method of invoking a service at a first machine, said method comprising the steps of:

receiving at said first machine a service invocation request;

invoking said service in response to said request; and

transmitting from said first machine a service invocation reply message in compliance with a markup language-based message encoding wherein each element in said message is associated with a type indicator selected from a group of names, said group of names including at least an array type name indicating that the corresponding element is an n-dimensional array containing a plurality of data items, where n is an integer and $n \geq 1$, said message including at least one data item which is a multi-level nested array element where each nesting level corresponds to a respective dimension of said array element; and

transmitting said service invocation reply message from said second machine.

93. A method of invoking a service at a first machine, said method comprising the steps of:

transmitting a service invocation request from a second machine; and

receiving at said second machine a service invocation reply message in compliance with a markup language-based message encoding wherein each element in said message is associated with a type indicator selected from a group of names, said group of names including at least an array type name indicating that the corresponding element is an n-dimensional array containing a plurality of data items, where n is an integer and $n \geq 1$, said message including at least one data item which is a multi-level nested array element where each nesting level corresponds to a respective dimension of said array element.

94. A method of invoking a service at a first machine from a second machine, said method comprising the steps of:

generating a service invocation request message at said second machine in compliance with a markup language-based message encoding wherein each element in said message is associated with a type indicator selected from a group of names, said group of names including at least an array type name indicating that the corresponding element is an n-dimensional array containing a plurality of data items, where n is an integer and $n \geq 1$, said request message including at least one data item which is an array of dimension n and a label associated with said data item and designating said data item as having an array type, said label indicating a value of n but not indicating a size for each of said n dimensions; and

transmitting said service invocation request message from said second machine.

95. A method of invoking a service at a first machine, comprising the steps of:

receiving at said first machine a service invocation request message generated at a second machine in compliance with a markup language-based message encoding wherein each element in said message is associated with a type indicator selected from a group of names, said group of names including at least an array type name indicating that the corresponding element is an n-

dimensional array containing a plurality of data items, where n is an integer and $n \geq 1$, said request message including at least one data item which is an array of dimension n and a label associated with said data item and designating said data item as having an array type, said label indicating a value of n but not indicating a size for each of said n dimensions; and
invoking said service in response to said message.

96. A method of invoking a service at a first machine, said method comprising the steps of:

receiving at said first machine a service invocation request;
invoking said service in response to said request; and
transmitting from said first machine a service invocation reply message in compliance with a markup language-based message encoding wherein each element in said message is associated with a type indicator selected from a group of names, said group of names including at least an array type name indicating that the corresponding element is an n -dimensional array containing a plurality of data items, where n is an integer and $n \geq 1$, said reply message including at least one data item which is an array of dimension n and a label associated with said data item and designating said data item as having an array type, said label indicating a value of n but not indicating a size for each of said n dimensions; and
transmitting said service invocation reply message from said second machine.

97. A method of invoking a service at a first machine, said method comprising the steps of:

transmitting a service invocation request from a second machine; and
receiving at said second machine a service invocation reply message in compliance with a markup language-based message encoding wherein each element in said message is associated with a type indicator selected from a group of names, said group of names including at least an array type name

indicating that the corresponding element is an n-dimensional array containing a plurality of data items, where n is an integer and $n \geq 1$, said reply message including at least one data item which is an array of dimension n and a label associated with said data item and designating said data item as having an array type, said label indicating a value of n but not indicating a size for each of said n dimensions.

98. A method of invoking a service at a first machine from a second machine, said method comprising the steps of:

generating a service invocation request message at said second machine in compliance with a mark-up language-based message encoding wherein each element in said message is associated with a type indicator selected from a group of names, said group of names including at least an array type name indicating that the corresponding element is an n-dimensional array containing a plurality of data items, where n is an integer and $n \geq 1$, said request message including at least one data item which is an array of dimension n and a label associated with said data item and designating said data item as having an array type, said encoding requiring that all data items contained within said array as direct children have the same type as one another; and

transmitting said service invocation request message from said second machine.

99. A method of invoking a service at a first machine, comprising the steps of:

receiving at said first machine a service invocation request message generated at a second machine in compliance with a markup language-based message encoding wherein each element in said message is associated with a type indicator selected from a group of names, said group of names including at least an array type name indicating that the corresponding element is an n-dimensional array containing a plurality of data items, where n is an integer and $n \geq 1$, said request message including at least one data item which is an

array of dimension n and a label associated with said data item and designating said data item as having an array type, said encoding requiring that all data items contained within said array as direct children have the same type as one another; and

invoking said service in response to said message.

100. A method of invoking a service at a first machine, said method comprising the steps of:

receiving at said first machine a service invocation request;

invoking said service in response to said request; and

transmitting from said first machine a service invocation reply message in compliance with a markup language-based message encoding wherein each element in said message is associated with a type indicator selected from a group of names, said group of names including at least an array type name indicating that the corresponding element is an n -dimensional array containing a plurality of data items, where n is an integer and $n \geq 1$, said reply message including at least one data item which is an array of dimension n and a label associated with said data item and designating said data item as having an array type, said encoding requiring that all data items contained within said array as direct children have the same type as one another; and

transmitting said service invocation reply message from said second machine.

101. A method of invoking a service at a first machine, said method comprising the steps of:

transmitting a service invocation request from a second machine; and

receiving at said second machine a service invocation reply message in compliance with a markup language-based message encoding wherein each element in said message is associated with a type indicator selected from a group of names, said group of names including at least an array type name indicating that the corresponding element is an n -dimensional array containing

a plurality of data items, where n is an integer and $n \geq 1$, said reply message including at least one data item which is an array of dimension n and a label associated with said data item and designating said data item as having an array type, said encoding requiring that all data items contained within said array as direct children have the same type as one another.

102. A method according to any one of claims 98-101, wherein said label identifies said same type.

103. A method according to any one of claims 94-101, wherein said label is expressed as an XML attribute of said element such that the dimension n is given by the value of the attribute

104. A method according to any one of claims 94-101, wherein said message is an XML document.

105. A method according to any one of claims 90-93, wherein said message includes a label associated with said data item and designating said data item as having an array type.

a

106. A method according to claim 105, wherein each of said second array elements includes at least one data item, with all data items in each of said second array elements being of the same type as one another.

107. A method according to claim 106, wherein said label indicates the type associated with all data items contained in said array.

108. A method according to any one of claims 94-101, wherein said label indicates a value of n but does not indicate a size for each of said n dimensions.

109. A method according to any one of claims 90-93, wherein said message includes a label associated with said data item and designating said data item as having an array type, said encoding requiring that all data items contained within said array as direct children have the same type as one another.

110. A method of invoking a service at a first machine from a second machine, said method comprising the steps of:

generating a service invocation request message at said second machine in compliance with a markup language-based message encoding, wherein each element in said message is associated with an element type name selected from a group of names including at least first and second element type names (e.g., RECORD and OBJECT), wherein said message associates an element of said first type name with an ID value, and wherein said message includes an element of said second type name (OBJECT) which specifies said ID value; and

transmitting said service invocation request message from said second machine.

111. A method of invoking a service at a first machine, comprising the steps of:

receiving at said first machine a service invocation request message generated at a second machine in compliance with a markup language-based message encoding, wherein each element in said message is associated with an element type name selected from a group of names including at least first and second element type names (e.g., RECORD and OBJECT), wherein said message associates an element of said first type name with an ID value, and wherein said message includes an element of said second type name (OBJECT) which specifies said ID value; and

invoking said service in response to said message.

112. A method of invoking a service at a first machine, said method comprising the steps of:

receiving at said first machine a service invocation request;

invoking said service in response to said request; and

transmitting from said first machine a service invocation reply message in compliance with a markup language-based message encoding, wherein each element in said message is associated with an element type name selected from a group of names including at least first and second element type names (e.g., RECORD and OBJECT), wherein said message associates an element of said first type name with an ID value, and wherein said message includes an element of said second type name (OBJECT) which specifies said ID value; and

transmitting said service invocation reply message from said second machine.

113. A method of invoking a service at a first machine, said method comprising the steps of:

transmitting a service invocation request from a second machine; and

receiving at said second machine a service invocation reply message in compliance with a markup language-based message encoding, wherein each element in said message is associated with an element type name selected from a group of names including at least first and second element type names (e.g., RECORD and OBJECT), wherein said message associates an element of said first type name with an ID value, and wherein said message includes an element of said second type name (OBJECT) which specifies said ID value.

114. A method of invoking a service at a first machine from a second machine, said method comprising the steps of:

generating a service invocation request message at said second machine in compliance with a markup language-based message encoding, wherein each element in said message is associated with an element type name selected from a group of names, said group of names including at least one placeholder

receiving at said second machine a service invocation reply message in compliance with a markup language-based message encoding, wherein , wherein each element in said message is associated with an element type name selected from a group of names, said group of names including at least one placeholder element type name (e.g., NULL or OBJECT) that designates a placeholder element which represents the absence of data.

118. A method according to any one of claims 114-117, wherein said placeholder element (e.g., NULL) represents a programming language null object reference.

119. A method according to any one of claims 114-117, wherein said placeholder element (e.g., OBJECT) identifies a data item contained elsewhere in said message.

120. A method according to any one of claims 110-113, wherein said message includes a type label associated with said placeholder element.

121. A method according to any one of claims 110-113, wherein said message includes a semantic label associated with said placeholder element.

122. A method according to claim 120, wherein said message includes a semantic label associated with said placeholder element.

123. A method according to any one of claims 110-113, wherein said encoding permits any data item in a message to be associated with an ID which uniquely identifies said data item within said message.

124. A method according to claim 123, wherein said ID is associated with a data item via an XML attribute on said data item whose value is said ID.